Journal of Computational and Applied Mathematics 235 (2011) 5357–5366

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

A parallel well-balanced finite volume method for shallow water equations with topography on the cubed-sphere

Chao Yang [a,*], Xiao-Chuan Cai [b]

[a] Institute of Software, Chinese Academy of Sciences, Beijing 100190, PR China

[b] Department of Computer Science, University of Colorado at Boulder, Boulder, CO 80309, USA

ARTICLE INFO

MSC:
86A10
76M12
35L65
65Y05

Keywords:
Shallow water equations
Topography
Well-balanced scheme
Exact-C-property
Cubed-sphere
Parallel scalability

ABSTRACT

A finite volume scheme for the global shallow water model on the cubed-sphere mesh is proposed and studied in this paper. The new cell-centered scheme is based on Osher's Riemann solver together with a high-order spatial reconstruction. On each patch interface of the cubed-sphere only one layer of ghost cells is needed in the scheme and the numerical flux is calculated symmetrically across the interface to ensure the numerical conservation of total mass. The discretization of the topographic term in the equation is properly modified in a well-balanced manner to suppress spurious oscillations when the bottom topography is non-smooth. Numerical results for several test cases including a steady-state nonlinear geostrophic flow and a zonal flow over an isolated mountain are provided to show the flexibility of the scheme. Some parallel implementation details as well as some performance results on a parallel supercomputer with more than one thousand processor cores are also provided.

1. Introduction

Numerical simulation of shallow water flows on a rotating sphere plays an important role in atmospheric sciences. The intrinsic curvature properties of the spherical geometry lead to numerical difficulties for the choice of the computational mesh aiming at uniformity, non-singularity and continuity. Great successes have been made during the past several decades in solving shallow water equations using the traditional latitude–longitude (Lat–Lon) mesh. However, the pole-singularity as well as the high non-uniformity of the Lat–Lon mesh become the two main disadvantages especially when the mesh is gradually refined for large scale parallel simulations. To circumvent these two problems, a variety of meshes have been proposed, studied, or revisited, most of which are composite meshes consisting of several patches connected or overlapped together to cover the whole sphere, e.g., the icosahedron geodesic mesh [1,2], the cubed-sphere mesh [3], the Yin–Yang mesh [4], among others.

In the present study we focus on the gnomonic cubed-sphere mesh [3,5], which is generated by mapping the six faces of the inscribed cube to the sphere surface using the gnomonic projection. The six expanded patches are continuously attached together with proper boundary conditions. However, the mesh lines on different patches are not smoothly connected thus extra numerical treatments are required on patch interfaces. Several efforts have been made on constructing discretization schemes for shallow water equations on the cubed-sphere, such as the finite volume method [6–8], the spectral element method [9,10], the discontinuous Galerkin method [11–13], among others. As is well known, the numerical conservation of integral invariants is crucial for atmospheric simulations. When a cell-centered finite volume method is used, however, interpolated function values on ghost cells are needed across patch boundaries thus the mass conservation could be violated

* Corresponding author.
E-mail addresses: yangchao@iscas.ac.cn (C. Yang), cai@cs.colorado.edu (X.-C. Cai).

0377-0427/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2011.01.016

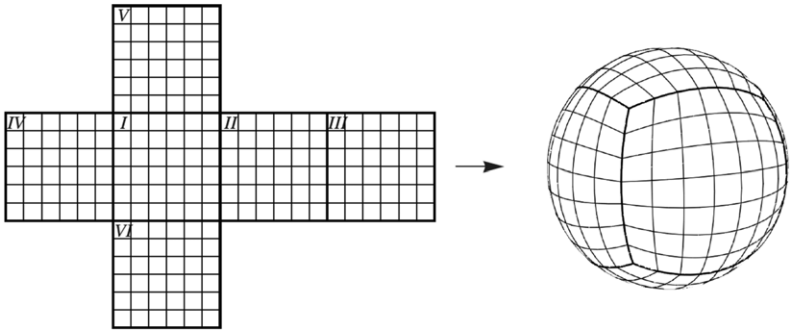

Fig. 1. The cubed-sphere mesh can be obtained by mapping the six faces of an inscribed cube covered with uniform meshes to the sphere surface. Each mesh line of the cubed-sphere is on a great circle.

despite the fact that the numerical method is locally conservative; see [6] for example. In this paper, we propose a second-order finite volume scheme where only one layer of ghost cells is needed and the numerical flux is calculated symmetrically across the interface to ensure the numerical conservation of the total mass.

When a variable bottom topography is involved in shallow water equations, special attention should be given to the discretization of the topographic source term. The scheme must satisfy a well-balanced property between the numerical flux and the source term, so that stationary or almost stationary solutions can be captured, and more importantly, spurious oscillations can be suppressed near the non-smooth area of the topography. Some techniques such as the Strang's operator splitting are often used to deal with these problems [6]. However, extra splitting error is included in the scheme and the method is not natural to be incorporated with an implicit time integration. In our previous work [14] we studied a first-order cell-centered finite volume method using Osher's Riemann solver to solve shallow water equations without topography. One purpose of this paper is to propose and study a well-balanced discretization of the topographic source term with a second-order cell-centered finite volume scheme.

The remainder of this paper is organized as follows. In Section 2, we propose a finite volume scheme for shallow water equations on the cubed-sphere. Numerical treatments for the mass conservation and the topographic source terms are then discussed in Sections 3 and 4. We present some numerical results for several test cases to show the accuracy and the flexibility of the scheme in Section 5. Some parallel implementation details as well as some performance results are provided in the end of Section 5. The paper is then concluded in Section 6.

2. A well-balanced finite volume scheme on the cubed-sphere

Suppose $(x, y) \in [-\pi/4, \pi/4]^2$ are the local curvilinear coordinates on a patch of the gnomonic cubed-sphere (see Fig. 1), shallow water equations can be written in the following conservative form:

$$\frac{\partial Q}{\partial t} + \frac{1}{\Lambda}\frac{\partial(\Lambda F)}{\partial x} + \frac{1}{\Lambda}\frac{\partial(\Lambda G)}{\partial y} + S = 0, \tag{1}$$

with

$$Q = \begin{pmatrix} h \\ hu \\ hv \end{pmatrix}, \qquad F = \begin{pmatrix} hu \\ huu + \frac{1}{2}gg^{11}h^2 \\ huv + \frac{1}{2}gg^{12}h^2 \end{pmatrix}, \qquad G = \begin{pmatrix} hv \\ huv + \frac{1}{2}gg^{12}h^2 \\ hvv + \frac{1}{2}gg^{22}h^2 \end{pmatrix}, \qquad S = \begin{pmatrix} 0 \\ S_1 \\ S_2 \end{pmatrix},$$

and

$$S_1 = \Gamma^1_{11}(huu) + 2\Gamma^1_{12}(huv) + f\Lambda\left(g^{12}hu - g^{11}hv\right) + gh\left(g^{11}\frac{\partial b}{\partial x} + g^{12}\frac{\partial b}{\partial y}\right),$$

$$S_2 = 2\Gamma^2_{12}(huv) + \Gamma^2_{22}(hvv) + f\Lambda\left(g^{22}hu - g^{12}hv\right) + gh\left(g^{12}\frac{\partial b}{\partial x} + g^{22}\frac{\partial b}{\partial y}\right).$$

Here h is the thickness of the fluid (atmosphere), (u, v) are the contravariant components of the fluid velocity, g is the gravitational constant and f is the Coriolis parameter due to the rotation of the sphere. The bottom topography is b which describes the height of the spherical surface, thus the surface level of the fluid is $H = h + b$. The variable coefficients g^{mn}, Λ and Γ^{ℓ}_{mn} are only dependent on the curvilinear coordinates and their detailed expressions can be found in [14].

Let us denote the six patches of the cubed-sphere as $\mathcal{P}^k$, $k = 1, \ldots, 6$. Suppose $\mathcal{P}^k$ is covered by a logically rectangular $N \times N$ mesh, which is equally spaced in the computational domain $\{(x, y) \in [-\pi/4, \pi/4]^2\}$ with mesh size $\hbar = \pi/2N$.

Patch $\mathcal{P}^k$ is then divided into small mesh cells $\mathcal{C}_{ij}^k$ with cell centers (x_i, y_j), $i, j = 1, \ldots, N$. The approximate solution in cell $\mathcal{C}_{ij}^k$ at time t is defined as

$$Q_{ij}^k \approx \frac{1}{\hbar^2 \Lambda_{ij}^k} \int_{y_j-\hbar/2}^{y_j+\hbar/2} \int_{x_i-\hbar/2}^{x_i+\hbar/2} \Lambda(x, y) Q(x, y, t) \mathrm{d}x\mathrm{d}y. \tag{2}$$

Here Λ_{ij}^k is evaluated at the cell center of $\mathcal{C}_{ij}^k$. The superscript k is sometimes ignored for convenience since most expressions are identical on different patches.

After discretizing the shallow water system (1) using a cell-centered finite volume method, we obtain the following semi-discrete system:

$$\frac{\partial Q_{ij}}{\partial t} + \frac{(\Lambda F)_{i+\frac{1}{2},j} - (\Lambda F)_{i-\frac{1}{2},j}}{\Lambda_{ij}\hbar} + \frac{(\Lambda G)_{i,j+\frac{1}{2}} - (\Lambda G)_{i,j-\frac{1}{2}}}{\Lambda_{ij}\hbar} + S_{ij} = 0. \tag{3}$$

Here the numerical fluxes are evaluated on the four cell boundaries, e.g.,

$$(\Lambda F)_{i+\frac{1}{2},j} \approx \frac{1}{\hbar} \int_{y_j-\hbar/2}^{y_j+\hbar/2} \Lambda(x_i + \hbar/2, y) F(x_i + \hbar/2, y, t) \mathrm{d}y. \tag{4}$$

In the current study, the numerical fluxes are approximated using Osher's Riemann solver [15,16], i.e.,

$$(\Lambda F)_{i+\frac{1}{2},j} = \Lambda_{i+\frac{1}{2},j} F^{(o)}(Q^-_{i+\frac{1}{2},j}, Q^+_{i+\frac{1}{2},j}) = \Lambda_{i+\frac{1}{2},j} F(Q^*_{i+\frac{1}{2},j}), \tag{5}$$

where $Q^-_{i+\frac{1}{2},j}$, $Q^+_{i+\frac{1}{2},j}$ are the "left" and "right" constant states by taking limits on the reconstructed function of Q towards local cell boundaries. If no topography is involved in the equation, Osher's method leads to

$$h^* = \frac{1}{4gg^{11}} \left[\frac{1}{2} \left(u^- - u^+\right) + \sqrt{gg^{11}} \left(\sqrt{h^-} + \sqrt{h^+} \right) \right]^2, \tag{6}$$

$$u^* = \frac{1}{2}(u^- + u^+) + \sqrt{gg^{11}} \left(\sqrt{h^-} - \sqrt{h^+} \right), \tag{7}$$

$$v^* = \begin{cases} v^- + \dfrac{g^{12}}{g^{11}} \left(u^* - u^-\right), & \text{if } u^* \geq 0 \\ v^+ + \dfrac{g^{12}}{g^{11}} (u^* - u^+), & \text{otherwise,} \end{cases} \tag{8}$$

where we need $|u| < \sqrt{gg^{11}h}$. The calculation of G follows an analogous way; see [14] for details. Suppose Q_{ij}, $Q_{i\pm1,j}$, $Q_{i,j\pm1}$ are already known, we use the following piecewise linear method (PLM) to construct Q on cell $\mathcal{C}_{ij}$:

$$\begin{aligned} Q^-_{i+\frac{1}{2},j} &= Q_{i,j} + 0.5 \cdot \text{limiter}\left(Q_{i,j} - Q_{i-1,j}, Q_{i+1,j} - Q_{i,j}\right), \\ Q^+_{i-\frac{1}{2},j} &= Q_{i,j} - 0.5 \cdot \text{limiter}\left(Q_{i,j} - Q_{i-1,j}, Q_{i+1,j} - Q_{i,j}\right). \end{aligned} \tag{9}$$

In the current study, we simply use the following slope limiter

$$\text{limiter}(a, b) = \frac{a + b}{2}.$$

More complicated limiters can be incorporated to avoid spurious oscillations if the solution contains strong shocks.

Ghost cells are used to pass information between the six patches. The ghost cells can be obtained either directly from neighboring patches or by extending several layers of meshes on each patch [6]. Our boundary treatment which will be further discussed in the next subsection is a modification of the latter method because it is more accurate for vector-valued functions [5,6].

We use an explicit total variation-diminishing Runge–Kutta (TVD-RK) method [17] for the time integration. Given a semi-discrete system $\frac{\partial Q}{\partial t} + \mathcal{L}(Q) = 0$, the second-order TVD-RK method reads

$$\begin{aligned} \overline{Q}^{(m)} &= Q^{(m-1)} - \Delta t \mathcal{L}(Q^{(m-1)}), \\ Q^{(m)} &= \frac{1}{2} Q^{(m-1)} + \frac{1}{2} \overline{Q}^{(m)} - \frac{1}{2} \Delta t \mathcal{L}(\overline{Q}^{(m)}). \end{aligned} \tag{10}$$

The time step size is adaptively controlled so that the corresponding CourantFriedrichsLewy (CFL) number is fixed to 0.5. It can be verified (see Appendix) that the maximum eigenvalue of the flux Jacobians of (1) is $U = \max\{|u| + \sqrt{gg^{11}h}, |v| + \sqrt{gg^{22}h}\}$. Then the CFL number can be obtained via $\text{CFL} = U\Delta t/\hbar$.

3. Numerical conservation of the total mass

Denote the discrete integral operator on the cubed-sphere as

$$I(h) = \sum_{k=1}^{6} \sum_{i,j=1}^{N} (\Lambda_{ij} h_{ij}^k). \tag{11}$$

The mass conservation of (3) is then equivalent to

$$I\left(\frac{\partial h}{\partial t}\right) = \frac{1}{\hbar} \sum_{k=1}^{6} \sum_{i,j=1}^{N} [(\Lambda hu)^k_{i+\frac{1}{2},j} - (\Lambda hu)^k_{i-\frac{1}{2},j} + (\Lambda hv)^k_{i,j+\frac{1}{2}} - (\Lambda hv)^k_{i,j-\frac{1}{2}}] = 0. \tag{12}$$

It is straightforward to obtain numerical mass conservation if the PDE system is written in a conservative form as in (1). However, for the shallow water equations discretized on the cubed-sphere, the numerical flux for mass conversation across patch interface needs to be treated carefully. For example, on the interface Γ_{12} of patches I and II, the numerical flux for the mass conservation of each patch should remain equal, say,

$$(hu)^{\mathrm{I}}_{N+\frac{1}{2},j} = (hu)^{\mathrm{II}}_{\frac{1}{2},j}. \tag{13}$$

In order to do this, the two reconstructed constant states on Γ_{12} are

$$(Q^{\mathrm{I}}_{N+\frac{1}{2},j})^- = Q^{\mathrm{I}}_{N,j} + 0.5 \cdot \mathrm{limiter}(Q^{\mathrm{I}}_{N,j} - Q^{\mathrm{I}}_{N-1,j}, Q^{\mathrm{I}}_{N+1,j} - Q^{\mathrm{I}}_{N,j}), \tag{14}$$

$$(Q^{\mathrm{II}}_{\frac{1}{2},j})^+ = Q^{\mathrm{II}}_{0,j} + 0.5 \cdot \mathrm{limiter}(Q^{\mathrm{II}}_{0,j} - Q^{\mathrm{II}}_{-1,j}, Q^{\mathrm{II}}_{1,j} - Q^{\mathrm{II}}_{0,j}), \tag{15}$$

where $Q^{\mathrm{I}}_{N+1,j}$ and $Q^{\mathrm{II}}_{-1,j}$ are values defined on ghost cells by interpolating from the values on neighboring patches. From (5), Osher's flux on each patch is

$$F^{\mathrm{I}}_{N+\frac{1}{2},j} = F^{(o)}((Q^{\mathrm{I}}_{N+\frac{1}{2},j})^-, T_{\mathrm{I}}^{-1} T_{\mathrm{II}} (Q^{\mathrm{II}}_{\frac{1}{2},j})^+), \tag{16}$$

$$F^{\mathrm{II}}_{\frac{1}{2},j} = F^{(o)}(T_{\mathrm{II}}^{-1} T_{\mathrm{I}} (Q^{\mathrm{I}}_{N+\frac{1}{2},j})^-, (Q^{\mathrm{II}}_{\frac{1}{2},j})^+), \tag{17}$$

where T_{I} and T_{II} are transform matrices of $Q = (h, hu, hv)^T$ from local cubed-sphere coordinates of patch *I* and *II* into spherical coordinates, respectively. It can be calculated that

$$T_{\mathrm{I}}^{-1} T_{\mathrm{II}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \sin 2y & 1 \end{pmatrix}, \qquad T_{\mathrm{II}}^{-1} T_{\mathrm{I}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -\sin 2y & 1 \end{pmatrix}, \quad \text{on } \Gamma_{12}.$$

Insert (14)–(15) into (16)–(17) and use (6)–(8) we obtain (13). Different from the numerical treatment for boundary conditions in [6], only one layer of ghost cells is needed here and the numerical flux is calculated symmetrically across the interface to ensure the numerical conservation of total mass. The mass conservations on other patch interfaces can be accessed in an analogous way.

4. Numerical treatment for the bottom topographic terms

Some of the source terms in the expression of S involve bottom topography,

$$S_{T1} = gh\left(g^{11}\frac{\partial b}{\partial x} + g^{12}\frac{\partial b}{\partial y}\right), \qquad S_{T2} = gh\left(g^{12}\frac{\partial b}{\partial x} + g^{22}\frac{\partial b}{\partial y}\right).$$

The discretization of these terms should be carried out with special care. For example, a naive centered finite difference such as

$$(S_{T1})_{ij} = g(g^{11}h)_{ij}\frac{b_{i+1,j} - b_{i-1,j}}{2\hbar} + g(g^{12}h)_{ij}\frac{b_{i,j+1} - b_{i,j-1}}{2\hbar}$$

might lead to spurious oscillations near the non-smooth area of b.

For a shallow water problem involving bottom topography, it is essential that the numerical scheme is well-balanced (see, e.g., [18,19]). One particularly important condition is the "C-property" [20], which means a stationary state at rest ("water at rest"), i.e., the special solution

$$H \equiv h + b = \text{constant}, \qquad \mathbf{v} = 0,$$

should be preserved. Here $\mathbf{v}$ is the fluid velocity.

Proposition 4.1. *The water-at-rest solution is preserved if the following modifications of the scheme are made.*

1. *In the calculation of the numerical flux* $F(Q^*_{i\pm\frac{1}{2},j})$ *in* (5), *replace* (7) *with*

$$u^* = \frac{1}{2}(u^- + u^+) + \frac{\sqrt{gg^{11}}(H^- - H^+)}{\sqrt{h^-} + \sqrt{h^+}}. \tag{18}$$

The numerical flux $G(Q^*_{i,j\pm\frac{1}{2}})$ *is treated in a similar manner.*

2. *The topographic source term* S_{T1} *is discretized as*

$$\begin{aligned}(S_{T1})_{ij} = \frac{g}{4\Lambda_{ij}\hbar}\Big\{&[(g^{11}\Lambda)_{i+\frac{1}{2},j} + (g^{11}\Lambda)_{i-\frac{1}{2},j}](h^*_{i+\frac{1}{2},j} + h^*_{i-\frac{1}{2},j})(b^*_{i+\frac{1}{2},j} - b^*_{i-\frac{1}{2},j})\\ &+[(g^{12}\Lambda)_{i,j+\frac{1}{2}} + (g^{12}\Lambda)_{i,j-\frac{1}{2}}](h^*_{i,j+\frac{1}{2}} + h^*_{i,j-\frac{1}{2}})(b^*_{i,j+\frac{1}{2}} - b^*_{i,j-\frac{1}{2}})\\ &-[(g^{11}\Lambda)_{i+\frac{1}{2},j} - (g^{11}\Lambda)_{i-\frac{1}{2},j}][(h^*_{i+\frac{1}{2},j})^2 + (h^*_{i-\frac{1}{2},j})^2]\\ &-[(g^{12}\Lambda)_{i,j+\frac{1}{2}} - (g^{12}\Lambda)_{i,j-\frac{1}{2}}][(h^*_{i,j+\frac{1}{2}})^2 + (h^*_{i,j-\frac{1}{2}})^2]\Big\}.\end{aligned} \tag{19}$$

Here

$$b^* = H^* - h^*, \tag{20}$$

where H^* *is calculated using an analogous formula to* h^*. *For instance, on the left and right cell boundaries,*

$$H^* = \frac{1}{4gg^{11}}\left[\frac{1}{2}\left(u^- - u^+\right) + \sqrt{gg^{11}}(\sqrt{H^-} + \sqrt{H^+})\right]^2. \tag{21}$$

The topographic source term S_{T2} *follows an analogous approach.*

Proof. It suffices to prove that

$$\frac{(\Lambda F)_{i+\frac{1}{2},j} - (\Lambda F)_{i-\frac{1}{2},j}}{\Lambda_{ij}\hbar} + \frac{(\Lambda G)_{i,j+\frac{1}{2}} - (\Lambda G)_{i,j-\frac{1}{2}}}{\Lambda_{ij}\hbar} + S_{ij} = 0, \tag{22}$$

given the water-at-rest initial state

$$H_{i,j} \equiv h_{i,j} + b_{i,j} = c \text{ (constant)}, \qquad u_{i,j} = v_{i,j} = 0. \tag{23}$$

Without loss of generality, we only need to prove

$$(\Lambda h^* u^*)\Big|_{i-\frac{1}{2},j}^{i+\frac{1}{2},j} + (\Lambda h^* v^*)\Big|_{i,j-\frac{1}{2}}^{i,j+\frac{1}{2}} = 0, \tag{24}$$

$$\left(\Lambda h^* u^* u^* + \frac{1}{2}gg^{11}\Lambda(h^*)^2\right)\Big|_{i-\frac{1}{2},j}^{i+\frac{1}{2},j} + \left(\Lambda h^* u^* v^* + \frac{1}{2}gg^{12}\Lambda(h^*)^2\right)\Big|_{i,j-\frac{1}{2}}^{i,j+\frac{1}{2}} = -(\Lambda_{ij}\hbar)(S_1)_{ij}. \tag{25}$$

From (9), we have

$$H^\pm = h^\pm + b^\pm = c, \qquad u^\pm = v^\pm = 0, \quad \text{at any cell boundaries.}$$

Further from (18) and (8), we have

$$u^* = v^* = 0, \quad \text{at any cell boundaries,}$$

which implies (24). And from (21), we have

$$H^* = h^* + b^* = c, \quad \text{at any cell boundaries.}$$

Then replace b^* with $c - h^*$ in (19) and use the equality

$$(A + C)(B - D) + (A - C)(B + D) = 2(AB - CD),$$

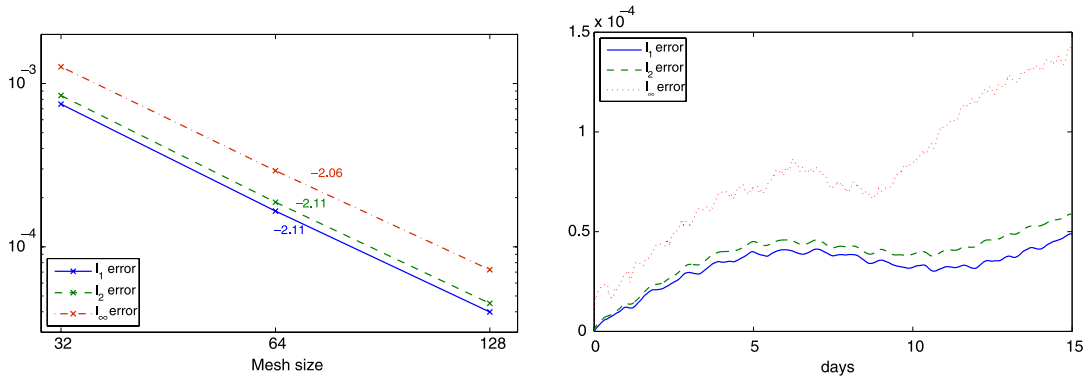

Fig. 2. Numerical results of the steady-state geostrophic flow problem ($\alpha = \pi/4$). Left figure: convergence of the height field errors at day 5. The orders of accuracy fitted using least squares are marked in the figure. Right figure: evolution history of the height field errors from day 0 to day 15 using a $128 \times 128 \times 6$ mesh.

we have

$$\begin{aligned}
-(\Lambda_{ij}\hbar)(S_1)_{ij} &= 0 - (\Lambda_{ij}\hbar)(S_{T_1})_{ij} \\
&= \frac{g}{4}\Big\{[(g^{11}\Lambda)_{i+\frac{1}{2},j} + (g^{11}\Lambda)_{i-\frac{1}{2},j}][(h^*_{i+\frac{1}{2},j})^2 - (h^*_{i-\frac{1}{2},j})^2] + [(g^{12}\Lambda)_{i,j+\frac{1}{2}} + (g^{12}\Lambda)_{i,j-\frac{1}{2}}] \\
&\quad \times [(h^*_{i,j+\frac{1}{2}})^2 - (h^*_{i,j-\frac{1}{2}})^2] + [(g^{11}\Lambda)_{i+\frac{1}{2},j} - (g^{11}\Lambda)_{i-\frac{1}{2},j}][(h^*_{i+\frac{1}{2},j})^2 + (h^*_{i-\frac{1}{2},j})^2] \\
&\quad + [(g^{12}\Lambda)_{i,j+\frac{1}{2}} - (g^{12}\Lambda)_{i,j-\frac{1}{2}}][(h^*_{i,j+\frac{1}{2}})^2 + (h^*_{i,j-\frac{1}{2}})^2]\Big\} \\
&= \left(\frac{1}{2}gg^{11}\Lambda(h^*)^2\right)\Big|^{i+\frac{1}{2}j}_{i-\frac{1}{2}j} + \left(\frac{1}{2}gg^{12}\Lambda(h^*)^2\right)\Big|^{i,j+\frac{1}{2}}_{i,j-\frac{1}{2}},
\end{aligned}$$

which implies (25). This completes the proof. □

5. Numerical results

5.1. A steady-state nonlinear geostrophic flow

The first test case is problem-2 taken from [21]. We use it to test the accuracy of our algorithm since the exact solution is known despite the fact that the topographic height is zero. Throughout all numerical tests, we use the characteristic time scale $\tau = 86\,400$ s (one day) and length scale $\rho = 6\,371\,220$ m (the radius of the Earth), thus the radius of the sphere is $a = 1.0\rho$, the rotation rate of the sphere is $\omega = 6.300288\tau^{-1}$ and the gravitational constant is $g = 11489.57\rho\tau^{-2}$. Denote the spherical coordinates on the sphere as (λ, θ), $\lambda \in [-\pi, \pi)$, $\theta \in [-\pi/2, \pi/2]$, then the exact solution is a steady-state flow as follows.

$$\begin{cases}
H = h_0 - \dfrac{1}{g}\left(a\omega u_0 + \dfrac{u_0^2}{2}\right)(\cos\alpha\sin\theta - \sin\alpha\cos\lambda\cos\theta)^2, \\
u_\lambda = u_0(\cos\alpha + \sin\alpha\cos\lambda\tan\theta)/a, \\
u_\theta = -u_0\sin\alpha\sin\lambda/a,
\end{cases}$$

where the reference height $h_0 = 2998.105$ m and the reference velocity $u_0 = (\pi/6)\rho\tau^{-1}$. We choose the flow orientation $\alpha = \pi/4$, then the Coriolis parameter is $f = \sqrt{2}\omega(\sin\theta - \cos\lambda\cos\theta)$.

Since the exact solution $Q = (h, hu, hv)^T$ is known, the relative error of the height field of numerical solution $\tilde{Q} = (\tilde{h}, \tilde{h}\tilde{u}, \tilde{h}\tilde{v})^T$ can be measured by

$$l_1 = I(|h - \tilde{h}|)/I(|h|), \qquad l_2 = \sqrt{I\left((h - \tilde{h})^2\right)/I\left(h^2\right)}, \qquad l_\infty = \max_{i,j,k}|h^k_{ij} - \tilde{h}^k_{ij}|/\max_{i,j,k}|h^k_{ij}|,$$

where I is the discrete integral operator defined by (11). Shown in the left figure of Fig. 2 are the convergence of the height field errors, which are similar to the second-order accuracy reported in [6]. The evolution history of the height field errors is provided in the right figure of Fig. 2.

Table 1
The relative errors of the height field at day 0.5, 128 × 128 × 6 mesh, water-at-rest flow.

l_1 error	l_2 error	l_∞ error
1.8243×10^{-17}	3.9123×10^{-17}	1.3878×10^{-16}

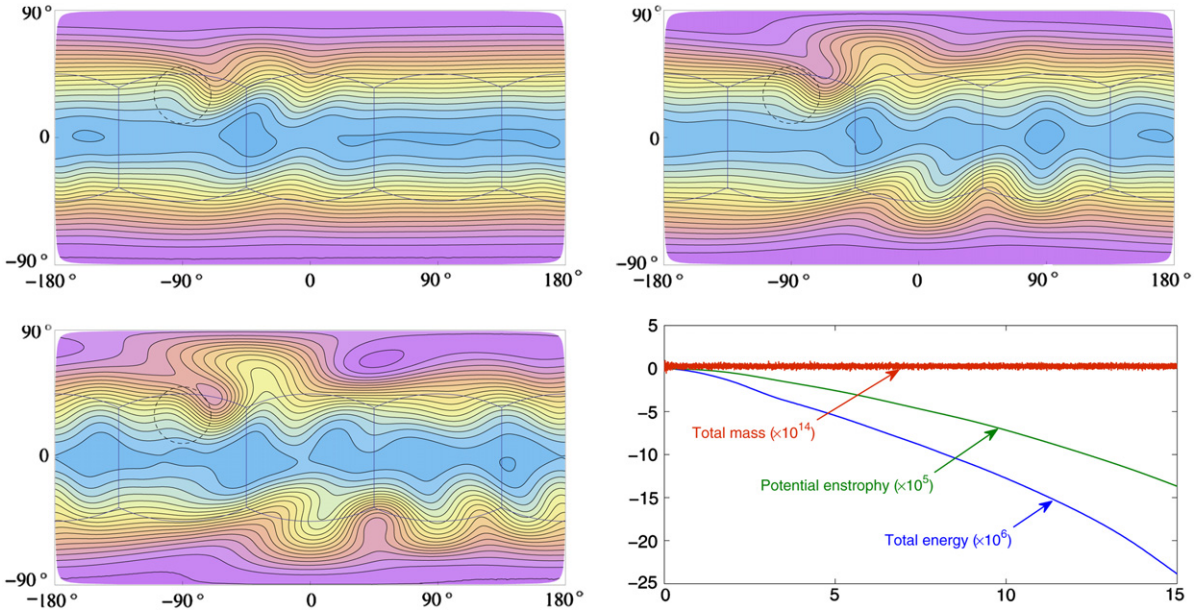

Fig. 3. Numerical results of the isolated mountain problem using a 128 × 128 × 6 mesh. The surface levels at days 5, 10, 15 are respectively plotted in the first three figures, where the mountain profile is indicated as the dotted lines, the contour levels are from 5000 to 5950 m with an interval of 50 m and the innermost lines near to the equator are at 5950 m. The conservation history is provided in the last figure.

5.2. Water-at-rest flow with random topography

To verify the "C-property" of the scheme, we consider a water-at-rest constant flow

$$H \equiv h + b = 0.1\rho, \quad \mathbf{v} = 0,$$

where the topographic height b is generated randomly between 0 to 0.05ρ. As predicted in Section 4, the numerical schemes preserve the steady-state solution exactly up to the machine precision. This fact can be observed in Table 1.

5.3. Zonal flow over an isolated mountain

This is problem-5 from [21], describing a zonal flow impinging an isolated mountain. It is modified from the steady-state geostrophic flow by adding a compactly supported, conical mountain to the domain. In this test the flow is purely zonal ($\alpha = 0$) with reference parameters $h_0 = 5960$ m and $u_0 = 20$ m s^{-1}. The mountain is centered at $(\lambda_c, \theta_c) = (-\pi/2, \pi/6)$ with height $b = b_0(1 - r/r_0)$, where $b_0 = 2000$ m, $r_0 = \pi/9$ and $r = \min\{r_0, \sqrt{(\lambda - \lambda_c)^2 + (\theta - \theta_c)^2}\}$. Note that the mountain height is not continuously differentiable on the boundary and at the center of the cone thus the topographic term has discontinuous coefficients. The discontinuities in the topographic term often result in spurious oscillations as observed in [22] if the numerical scheme is not well-balanced.

Since no analytical solution is available for this test, we compare our results with a reference solution using a spectral method with high resolution [23]. The numerical results after 5, 10, 15 days of integration are presented in the first three figures of Fig. 3. The results are in agreement with the reference solutions and no spurious oscillations are observed. The normalized conservation error at time t is measured as $\delta(\cdot) = [I(\cdot, t) - I(\cdot, 0)]/I(\cdot, 0)$. Here the integral invariants are the total mass $\delta(h)$, the total energy $\delta(E)$ and the potential enstrophy $\delta(\xi)$, where

$$E = \frac{h}{2}\mathbf{v}\cdot\mathbf{v} + \frac{g}{2}(H^2 - b^2) = \frac{\Lambda^2 h}{2}(g^{11}v^2 + g^{22}u^2 - 2g^{12}uv) + \frac{g}{2}(H^2 - b^2),$$

$$\xi = \frac{1}{2h}(\zeta + f)^2 = \frac{1}{2h}\left\{\frac{\partial[\Lambda^2(g^{11}v - g^{12}u)]}{\Lambda\partial x} - \frac{\partial[\Lambda^2(g^{22}u - g^{12}v)]}{\Lambda\partial y} + f\right\}^2,$$

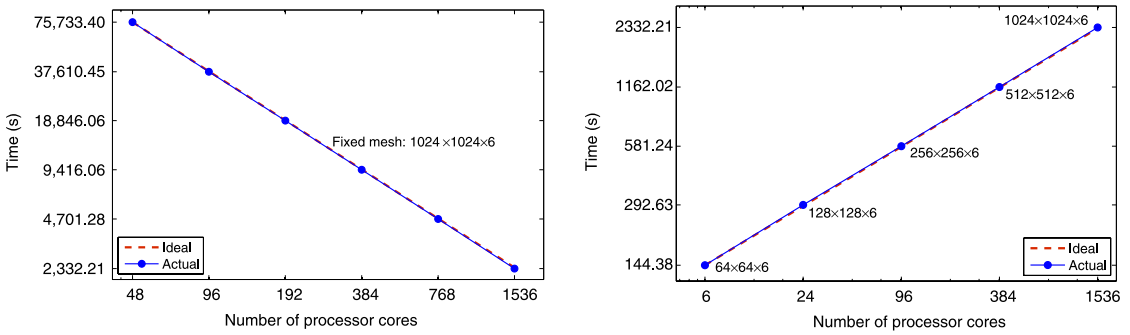

Fig. 4. Performance results of the isolated mountain problem at day 5 with Δt decided from CFL = 0.5. Left figure: strong scalability. Right figure: weak scalability.

where ζ is the relative vorticity. Drawn in the last figure of Fig. 3 are the normalized conservation errors $\delta(h)$, $\delta(E)$ and $\delta(\xi)$. It is evident that the total mass is conserved to the machine precision and the conservations of the other two values are also good.

5.4. Parallel implementation and performance tests

We implement our algorithm using the PETSc (Portable Extensible Toolkit for Scientific computation [24]) library. The portability of the solver is achieved through MPI, but most message-passing details are hidden inside PETSc and not explicitly needed in the application. The distributed array (DA) object in PETSc, which contains the parallel data layout information and communication information, is intended for use with logically rectangular meshes to optimize communication of vector components among distributed MPI processors. To take advantage of the DA object in PETSc, the six patches of the cubed-sphere can either be put together as a "thick" DA with $3 \times 6 = 18$ degrees of freedom per mesh point [14] or be one-to-one assigned with six DAs and only 3 degrees of freedom per mesh point for each patch. No significant difference is observed between the parallel performance of the two strategies. In the current study we choose to use the latter method because it is a more natural way to implement the cubed-sphere. So each of the six patch corresponds to one MPI group, which is handled by a DA object for parallel implementation. The six patches are then coupled with each other via boundary conditions on patch interfaces.

The performance tests are carried out on an IBM BlueGene/L supercomputer with 1024 dual-core processors. Each node has two IBM PowerPC 440 processors running at 700 MHz and 512 MB of memory. We run the isolated mountain test with different mesh sizes and different numbers of processor cores (np). For the strong scaling tests, we use a fixed mesh $1024 \times 1024 \times 6$ and in the ideal situation, the execution time should be saved by a factor of 2 as the number of processor cores doubles. As shown in the left figure of Fig. 4, the actual speedup is optimal and even slightly better than expected, e.g., the parallel efficiency for $np = 1536$ with respect to $np = 48$ is $75733.40/2332.21/32 = 101.48\%$. To further examine the scalability of our algorithm, we increase np as the mesh resolution is increased such that a consistent amount of unknowns per core is maintained. If the same final simulation time (such as day 5) is targeted, the time step size is halved for each doubling of resolution, thus the ideal execution time doubles. The weak scaling results are provided in the right figure of Fig. 4, revealing also an optimal scalability. Both the scaling results given in Fig. 4 are comparable to those reported in [25].

6. Conclusions and future work

A well-balanced method is proposed and studied for the shallow water equations on the cubed-sphere. The method is based on a cell-centered finite volume scheme with second-order accuracy. The numerical mass conservation is maintained due to the symmetrically calculated numerical fluxes on patch interfaces of the cubed-sphere. With a well-balanced discretization of the topographic source terms, the scheme is free of spurious oscillations coming from the non-smoothness of the topography. The parallel implementation is based on PETSc and the parallel scalability is optimal with up to 1536 processor cores.

As we can see from the weak scaling tests, although the per core degree of freedoms remains unchanged, the expected execution time doubles as the mesh is refined, due to the dependency between the time step size and the CFL number. A fully implicit time integration can be used to remove this dependency. In our previous work [14], a fully implicit domain decomposition method was studied for the first-order cell-centered finite volume scheme. At each time step, a nonlinear algebraic system needs to be solved. The nonlinear systems can be solved very efficiently by using the Newton–Krylov–Schwarz method [26] if a low-order spatial discretization is used, as reported in [14]. However, when we use a higher-order method such as the second-order finite volume scheme, the Jacobian matrix at each Newton iteration becomes highly ill-conditioned and very hard to solve. Besides, non-smoothness coming from the limiters of high-order discretization as well as from the topographic source terms challenges the convergence of the classical Newton's method.

How to construct efficient preconditioners for the Jacobian matrices obtained from high-order spatial discretizations and how to deal with the non-smoothness of the nonlinear function evaluations are the subjects of our future research.

Acknowledgements

CY was supported in part by NSFC under 10801125 and in part by 863 Program of China under 2010AA012301. XCC was supported in part by NSF grant CCF-0634894. The authors would like to thank the anonymous referees for their valuable suggestions to improve the presentation.

Appendix. Calculation of the flux Jacobians

The system of shallow water equations (1) can be rewritten as

$$\frac{\partial Q}{\partial t}+\frac{\partial F}{\partial x}+\frac{\partial G}{\partial y}+\left(\frac{F}{\Lambda}\frac{\partial \Lambda}{\partial x}+\frac{G}{\Lambda}\frac{\partial \Lambda}{\partial y}+S\right)=0. \tag{26}$$

Denote $Q=(h,hu,hv)^T=(q_1,q_2,q_3)^T$, we have

$$F=\left(q_2,q_2^2/q_1+gg^{11}q_1^2/2,q_2q_3/q_1+gg^{12}q_1^2/2\right)^T.$$

The flux Jacobian in the x-direction is

$$F'(Q)=\begin{pmatrix}0 & 1 & 0\\ -q_2^2/q_1^2+gg^{11}q_1 & 2q_2/q_1 & 0\\ -q_2q_3/q_1^2+gg^{12}q_1 & q_3/q_1 & q_2/q_1\end{pmatrix}=\begin{pmatrix}0 & 1 & 0\\ -u^2+gg^{11}h & 2u & 0\\ -uv+gg^{12}h & v & u\end{pmatrix}.$$

Solving the characteristic equation $|F'(Q)-\lambda^F I|=0$ leads to the eigenvalues of the flux Jacobian $F'(Q)$:

$$\lambda_1^F=u-\sqrt{gg^{11}h},\qquad \lambda_2^F=u,\qquad \lambda_3^F=u+\sqrt{gg^{11}h}.$$

An upper bound of the maximum eigenvalue of $F'(Q)$ is then given by

$$\lambda_M^F=|u|+\sqrt{gg^{11}h}.$$

Analogously the flux Jacobian in the y-direction can be calculated as

$$G'(Q)=\begin{pmatrix}0 & 0 & 1\\ -uv+gg^{12}h & v & u\\ -v^2+gg^{22}h & 0 & 2v\end{pmatrix},$$

with eigenvalues

$$\lambda_1^G=v-\sqrt{gg^{22}h},\qquad \lambda_2^G=v,\qquad \lambda_3^G=v+\sqrt{gg^{22}h},$$

and the maximum eigenvalue of $G'(Q)$ is bounded by

$$\lambda_M^G=|v|+\sqrt{gg^{22}h}.$$

References

[1] R. Sadourny, A. Arakawa, Y. Mintz, Integration of the nondivergent barotropic vorticity equation with an icosahedral–hexagonal grid for the sphere, Mon. Weather Rev. 96 (1968) 351–356.
[2] D.L. Williamson, Integration of the barotropic vorticity equation on a spherical geodesic grid, Tellus 20 (1968) 642–653.
[3] R. Sadourny, Conservative finite-difference approximations of the primitive equations on quasi-uniform spherical grids, Mon. Weather Rev. 100 (1972) 211–224.
[4] A. Kageyama, T. Sato, Yin-Yang grid: an overset grid in spherical geometry, Geochem. Geophys. Geosyst. 5 (2004).
[5] C. Ronchi, R. Iacono, P. Paolucci, The cubed-sphere: a new method for the solution of partial differential equations in spherical geometry, J. Comput. Phys. 124 (1996) 93–114.
[6] J.A. Rossmanith, A wave propagation method for hyperbolic systems on the sphere, J. Comput. Phys. 213 (2006) 629–658.
[7] W.M. Putman, S.-J. Lin, Finite-volumn transport on various cubed-sphere grids, J. Comput. Phys. 227 (2007) 55–78.
[8] C. Chen, F. Xiao, Shallow water model on cubed-sphere by multi-moment finite volume method, J. Comput. Phys. 227 (2008) 5019–5044.
[9] S.J. Thomas, R.D. Loft, Semi-implicit spectral element atmospheric model, J. Sci. Comput. 17 (2002) 229–350.
[10] J.M. Dennis, A. Fournier, W.F. Spotz, A. St.-Cyr., M.A. Taylor, S.J. Thomas, H. Tufo, High resolution mesh convergence properties and parallel efficiency of a spectral element atmospheric dynamical core, Int. J. High Perform. Comput. Appl. 19 (2005) 225–235.
[11] R.D. Nair, S.J. Thomas, R.D. Loft, A discontinuous Galerkin global shallow water model, Mon. Weather Rev. 133 (2005) 876–888.
[12] J.M. Dennis, M. Levy, R.D. Nair, H.M. Tufo, T. Voran, Towards an efficient and scalable discontinuous Galerkin atmospheric model, in: Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium, IPDPS'05, IEEE Computer Society, Washington, DC, USA, 2005, p. 257.1.
[13] J.M. Dennis, R.D. Nair, H.M. Tufo, M. Levy, T. Voran, Development of a scalable global discontinuous Galerkin atmospheric model, Int. J. Comput. Sci. Eng. (2006).
[14] C. Yang, J. Cao, X.-C. Cai, A fully implicit domain decomposition algorithm for shallow water equations on the cubed-sphere, SIAM J. Sci. Comput. 32 (2010) 418–438.

[15] S. Osher, F. Solomon, Upwind difference schemes for hyperbolic systems of conservation laws, Math. Comp. 38 (1982) 339–374.
[16] S. Osher, S. Chakravarthy, Upwind schemes and boundary conditions with applications to Euler equations in general geometries, J. Comput. Phys. 50 (1983) 447–481.
[17] S. Gottlieb, C.-W. Shu, E. Tadmore, Strong stability preserving high-order time integration methods, SIAM Rev. 43 (2001) 89–112.
[18] J.M. Greenberg, A.Y. Leroux, A well-balanced scheme for the numerical processing of source terms in hyperbolic equations, SIAM J. Numer. Anal. 33 (1996) 1–16.
[19] J.M. Gallardo, C. Parés, M. Castro, On a well-balanced high-order finite volume scheme for shallow water equations with topography and dry areas, J. Comput. Phys. 227 (2007) 574–601.
[20] A. Bermudez, M.E. Vazquez, Upwind methods for hyperbolic conservation laws with source terms, Comput. Fluids 23 (1994) 1049–1071.
[21] D.L. Williamson, J.B. Drake, J.J. Hack, R. Jakob, P.N. Swarztrauber, A standard test set for numerical approximations to the shallow water equations in spherical geometry, J. Comput. Phys. 102 (1992) 211–224.
[22] F.X. Giraldo, J.S. Hesthaven, T. Warburton, Nodal high-order discontinuous Galerkin methods for spherical shallow water equations, J. Comput. Phys. 181 (2002) 499–525.
[23] R. Jakob-Chien, J.J. Hack, D.L. Williamson, Spectral transform solutions to the shallow water test set, J. Comput. Phys. 119 (1995) 164–187.
[24] S. Balay, K. Buschelman, W.D. Gropp, D. Kaushik, M. Knepley, L.C. McInnes, B.F. Smith, H. Zhang, PETSc users manual, Argonne National Laboratory, 2010.
[25] W.M. Putman, Development of the finite-volume dynamical core on the cubed-sphere, Ph.D. Thesis, The Florida State University, 2007.
[26] X.-C. Cai, W.D. Gropp, D.E. Keyes, M.D. Tidriri, Newton–Krylov–Schwarz methods in CFD, in: R. Rannacher (Ed.), Notes in Numerical Fluid Mechanics: Proceedings of the Intl. Workshop on the Navier–Stokes Equations, Vieweg Verlag, Braunschweig, 1994, pp. 123–135.